

*b6*  
*cont.*

querying a control device having a plurality of human-actuated controls, said control device storing a genre descriptor indicating actions to be performed by application programs in said particular application program genre in response to said human-actuated controls;

obtaining, in response to said querying act, said genre descriptor; and

generating input to said application program in accordance with said genre descriptor.

---

45. (Amended) A method of enabling the use of an application program that executes on a computing device with a control device having human-actuated controls, said method comprising the acts of:

defining a plurality of application program genres, each of said genres comprising a set of semantics and not including commands interpretable by application programs classifiable in the genre;

*b7*

creating a genre descriptor, said genre descriptor indicating, for each one of said plurality of application program genres, actions to be performed by application programs in the respective application program genres in response to said human-actuated controls;

storing said genre descriptor in a memory of said control device, said memory being communicatively coupleable to said computing device whereby said genre descriptor is accessible to said computing device.

---

**REMARKS**

Claims 1-53 are pending in this application. As a result of the March 26, 2002 Office Action, all claims stand rejected. Independent claims 1, 3, 14, 23, 32, 36, and 45 have been amended.

The Examiner has rejected independent claim 1 under 35 U.S.C. § 103(a) as being unpatentable over U.S. Patent No. 6,085,265 to Kou in view of U.S. Patent No.

5,442,376 to Tannenbaum and U.S. Patent No. 5,157,384 to Greanias (which is incorporated by reference in Tannenbaum). The Examiner has further rejected independent claims 14, 23, and 26 under section 103(a) as being unpatentable over Tannenbaum and Greanias. Moreover, the Examiner has rejected all of the dependent claims as being unpatentable under section 103(a) based on various combinations of Kou, Tannenbaum, and Greanias. In response to these rejections, applicants have amended the independent claims to more particularly point out the invention. Applicants submit that these amendments define over the art cited by the Examiner. Moreover, since all independent claims, as amended, recite features not taught or suggested by the prior art, applicants submit that the dependent claims are patentable, at least by reason of their dependency.

#### The Prior Art Cited by the Examiner

The Examiner's 103(a) rejections of the claims rely on various combinations of Kou, Greanias, and Tannenbaum. Kou is cited for it's asserted teaching of Universal Serial Bus (USB) devices. Greanias and Tannenbaum are cited for various features, including the "genres" recited in the claims. Since the amendment serves to more particularly define the structure of a "genre," and since applicants' position is that this genre structure is sufficient to define over the prior art, the discussion below focuses on the portions of Greanias and Tannenbaum that the Examiner has cited as teaching a "genre." In particular, the Examiner has read the claim feature of a "genre" onto the "user profile" taught in Greanias and Tannenbaum.

#### Greanias

Greanias teaches a system for using input devices with applications. The computer system on which the applications run includes: an operating system that keeps track of which application is "active" (i.e., which one of several applications the user is currently using) (col. 5, ll. 1-15); an "advanced user interface" (AUI) that correlates input messages with commands that the application recognizes (col. 9, ll. 15-22); one or more input devices (col. 4, ll. 64-67); and an "environment link" that determines which

B

application was “active” at the time that an input message was generated, and resolves conflicts among contradictory information within the AUI (col. 8, ll. 56-59; col. 9, ll. 1-5).

When a user operates an input device (e.g., pressing a key on a keyboard, moving a mouse, etc.), a “recognition unit” associated with the particular device generates an “input token” (which Greanias also calls an “input message”) based on the raw event data captured from the device. For example, a series of mouse movements might be interpreted as drawing a circle, in which case the mouse’s recognition unit may generate a “circle” token. (Col. 6, l. 38 through col. 7, l. 16).

The AUI is used to convert input messages into commands that can be recognized and processed by an application. The AUI comprises an “interface profile module,” (col. 7, l. 66), which, in turn, comprises two types of components: an “application profile” and a “user profile” (col. 7, ll. 66-68; col. 9, ll. 14-17). The application profile and user profile are similar in the sense that “[b]oth types of profile associate a list lists [sic] of input messages with a list of corresponding commands ....” (col. 9, ll. 17-19). However, an application interface profile is associated with a particular application, whereas a user interface profile is associated with a particular user (col. 9, ll. 42-54; col. 10, ll. 5-21). For example, an application interface profile may state that a “circle” token corresponds to command “X” in that particular application. A user, however, may wish the “circle” token to be interpreted as command “Y” for all of the applications that the user frequently uses. Thus, the correspondence between the circle token and command “Y” can be stated in the user interface profile. A priority may be defined between application interface profiles and user interface profiles, and, where there is a conflict between them (e.g., two different commands corresponding to the “circle” token), the environment link resolves the conflict by determining whether the user profile or application profile has a higher priority (col. 8, l. 65 through col. 9, l. 7).

The application profile and user profile are files that correlate input messages with data that is usable by an application program. In particular, Greanias states:



The interface profile module 104 is comprised of sets of application profiles 105 and the user profile 107, which are files which list input messages produced by the AIS [Alternative Input Subsystem] 103 from the input signals received by input devices 36, 38, 40, mapped to keyboard, mouse or other commands which are usable by existing application programs, e.g., mouse clicks, keystroke messages, MACROs, utility programs, etc.

[Greanias, col. 7, l. 66 through col. 8, l. 5.] In other words, Greanias teaches that both the application profile and user profile correlate: (1) input messages received from input devices, with (2) commands that an application program can interpret as input. As explained below, the application program genres recited in the claims do not contain this type of correlation.

Tannenbaum

Tannenbaum is directed to a “method and system to recognize input events from a plurality of input devices.” (Tannenbaum, Abstract.) Tannenbaum teaches a “recognition provider,” which are adapted to recognize input for a particular type of device and to pass that input to a recognition subsystem. (Tannenbaum, col. 6, ll. 7-21.)

Tannenbaum incorporates Greanias by reference, (Tannenbaum, col. 1, ll. 35-37), and relies on Greanias for its teachings regarding interface profiles. (Tannenbaum, col. 6, ll. 22-25.) Additionally, Tannenbaum states that “the profiles link the registration ID numbers of the input event to a corresponding command which will be understood by the target object.” (Tannenbaum, col. 8, ll. 18-20.)

As discussed below, the claimed “application program genres” do not contain the type of correlation taught in Greanias.

The Claims of the Present Application, as Amended

Independent claims 1, 3, 14, 23, 32, 36, and 45 have been amended more particularly recite the structure of a “genre” as comprising “a set of semantics” and not including “commands interpretable by the application programs.” The Examiner has read the “application program genres” of claims these independent claims onto Greanias’s and

Tannenbaum's teachings of a "user profile." (See Office Action, p. 2-9, 12-15 ¶¶ 3-6, 8, 11, 12.) As discussed above, the user profile taught in Greanias and Tannenbaum contain lists that correlate input messages with commands that can be received and interpreted by an application program, which do not meet the genres recited in the independent claims, as amended.

As described in the present application, "[a] genre description for a specific genre defines mappings between input device controls and actions to be performed in a game of the genre. These actions are defined in terms of semantics or labels." (Application, p. 9, ll. 20-23.) Genres, however, need not contain actual commands that can be fed as input to an application program and interpreted by such a program. In fact, the Background section of the present application describes the drawbacks of prior art systems that contain such features:

Some manufacturers of devices with new features provide filters to accommodate existing applications; frequently, these filters simulate keyboard presses or mouse movements for games that do not recognize enhanced features of the new device. However, these *ad hoc* approaches are error-prone, may result in a relationship between device controls and software actions that feels unnatural to the user, and can only provide support for applications the device manufacturer knows about and chooses to support.

[Application, p. 2, l. 20 through p. 3, l. 4.] In other words, the present application describes the prior art as having the disadvantage that it relies on direct translation between the controls on a device and the input lexicon that an application recognizes. (In the above example, it is assumed that "keyboard presses or mouse movements" are the input that an application program understands.) Thus, the present invention seeks to improve over the prior art by providing a way to connect input devices with software in a way that is not dependent on the actual lexicon that an application recognizes. The "Examples" section on pages 24-31 of the application show examples of genres that have semantics, but do not include the command lexicon for a particular application. This genre structure creates greater flexibility, since the genre is not tied to the input vocabulary of any particular application. Correlating the semantics with the command

B

lexicon of the application program is the function of the A-S Correlation 231 (Application, FIG. 4). The A-S correlation, however, is not the claimed genre.

In accordance with the description of genre in the application, the independent claims have been amended to recite that the genres comprise semantics but do not include commands that are interpretable by the application program. The claimed structure of a genre is different from the “user profile” taught in Greanias and Tannenbaum. The “user profile” correlates input messages with commands that an application program interprets, whereas the claimed genre does not include commands interpretable by the application program. Additionally, the claimed genre comprises “semantics,” but the prior art “user profile” does not. Moreover, we have found no other portions of Kou, Greanias, or Tannenbaum that teach, suggest, or motivate a skilled artisan to create the feature of a genre that includes semantics and does not include commands interpretable by an application program

Accordingly, the independent claims, as amended, define over the prior art Kou, Greanias, and Tannenbaum patents, and are thus patentable. Additionally, dependent claims 2, 4-13, 15-23, 24-31, 33-35, 37-44, and 46-53 are patentable over Kou, Greanias, and Tannenbaum at least by reason of their dependency.

#### No New Matter

As noted above, the structure of a genre as comprising semantics and not including commands interpretable by an application program is supported by Examples contained in the detailed description at pages 24-31. Additionally, the feature of a genre as not including commands interpretable by an application program is supported by the Background section (p. 2-3), wherein it notes that a filter that translates directly into the input lexicon of an application program is a disadvantageous prior art technique.

#### CONCLUSION

For all the foregoing reasons, applicants submit that the claims, as amended, are patentable. Applicants thus request reconsideration of the Office Action, and that an early Notice of Allowance issue forthwith.

B

Respectfully submitted,



Peter M. Ullman  
Registration No. 43,963

Date: 7-26-02

WOODCOCK WASHBURN LLP  
One Liberty Place – 46<sup>th</sup> Floor  
Philadelphia, PA 19103  
Tel: (215) 568-3100  
Fax: (215) 568-3439

B

**VERSION WITH MARKINGS TO SHOW CHANGES MADE**

1. (Amended) A game control device that conforms to Universal Serial Bus (USB) device class definitions for Human Interface Devices (HIDs), comprising:

a plurality of human-actuated controls;

one or more HID descriptors that describe aspects of the human-actuated controls, the HID descriptors associating HID string indexes with the respective human-actuated controls;

control mappings corresponding to a plurality of application program genres, the control mappings indicating actions to be performed in application programs of particular genres in response to respective ones of the human-actuated controls, wherein the control mappings identify controls by their HID string indexes, each of said genres comprising a set of semantics and not including commands interpretable by the application programs.

3. (Amended) A computer peripheral comprising:

a plurality of human-actuated controls;

non-volatile memory containing control mappings corresponding to a plurality of application program genres, the control mappings indicating actions to be performed in application programs of particular genres in response to respective ones of the human-actuated controls, each of said genres comprising a set of semantics and not including commands interpretable by the application programs.

14. (Amended) A method comprising:

defining a plurality of application program genres, each of said genres comprising a set of semantics and not including commands interpretable by application programs classifiable in the genre;

running an application program that has been classified as a particular application program genre, wherein the application program is responsive to a plurality of human-actuated controls on a control device;

B

querying the control device to obtain a genre descriptor, the genre descriptor indicating actions to be performed by an application program of said particular application program genre in response to respective ones of the human-actuated controls.

23. (Amended) A computer-readable storage medium containing system services utilized by an application program to interact with a control device having a plurality of human-actuated controls, wherein the system services perform acts comprising:

receiving a request from an application program for a genre description corresponding to one of a plurality of application program genres, each of said genres comprising a set of semantics and not including commands interpretable by application programs classifiable in the genre;

querying the control device to obtain a genre descriptor, the genre descriptor indicating actions to be performed by an application program of said one of a plurality of application program genres in response to respective ones of the human-actuated controls;

returning the obtained genre descriptor to the requesting application program.

32. (Amended) A data transmission medium carrying a data structure comprising:

a header section indicating the number of human-actuated controls on a computer peripheral and the number of application program genres for which control mappings exist in the data structure;

a control section indicating HID string indexes for the respective controls on the computer peripheral;

a genre section indicating control mappings for the respective application program genres, each of said genres comprising a set of semantics and not including commands interpretable by application programs classifiable in the genre.

B

36. (Amended) A method of using an input device connected to a computing device with software executable on said computing device, said method comprising the acts of:

running an application program which is responsive to input, said application program being classified as a particular one of a plurality of application program genres, each of said genres comprising a set of semantics and not including commands interpretable by application programs classifiable in the genre;

querying a control device having a plurality of human-actuated controls, said control device storing a genre descriptor indicating actions to be performed by application programs in said particular application program genre in response to said human-actuated controls;

obtaining, in response to said querying act, said genre descriptor; and

generating input to said application program in accordance with said genre descriptor.

45. (Amended) A method of enabling the use of an application program that executes on a computing device with a control device having human-actuated controls, said method comprising the acts of:

defining a plurality of application program genres, each of said genres comprising a set of semantics and not including commands interpretable by application programs classifiable in the genre;

creating a genre descriptor, said genre descriptor indicating, for each one of said plurality of application program genres, actions to be performed by application programs in the respective application program genres in response to said human-actuated controls;

storing said genre descriptor in a memory of said control device, said memory being communicatively coupleable to said computing device whereby said genre descriptor is accessible to said computing device.

B